



UNIVERSIDAD TÉCNICA  
FEDERICO SANTA MARÍA

DEPARTAMENTO  
DE INFORMÁTICA

# Tarea 1

[github](#)

Autores:

José Campos

Matías Sandoval

Septiembre 2025

# 1- Validación

- a) ¿Como se ejecutará el programa?
  - El programa se ejecutará de manera local mediante consola
- b) ¿Dónde quedaran almacenados los datos?
  - Quedarán almacenados en una base de datos MySQL
- c) ¿Cómo se comportan las variables (que tipo son string, int, boolean, etc)
  - Los campos nombre, descripción, usuario, contraseña y categoría son strings, fecha está en formato date, y precio de entrada y cupos disponibles son enteros.
- d) ¿Cuál será el formato de el usuario y las contraseñas?
  - El usuario tiene un máximo de 50 caracteres mientras que las contraseñas tienen un máximo de 255 caracteres
- e) ¿Existe algún parámetro que pueda ser nulo?
  - Solo las descripciones del evento pueden ser nulas
- f) ¿Qué ocurre cuando se quiere rembolsar una entrada y ya ocurrió el evento?
  - El usuario, si está autenticado, tiene la capacidad de registrar la compra o la devolución según le parezca pertinente.
- g) ¿Cuándo genero el reporte que eventos me mostrará?
  - Muestra un resumen de los eventos, considerando la cantidad de ellos disponibles, cupos totales y eventos sin cupos. Solo se consideran los eventos que no hayan ocurrido aún.
- h) ¿Cómo se muestran los filtros?
  - Se mostrarán en cascada, y se filtra según: nombre, categoría, fecha y precio, mostrando todos los eventos que cumplan con aquellas características

Se añadió un id único a los eventos para poder mejorar su manejo.

# 2- Verificación

Para la verificación de los requisitos se realizaron pruebas a medida que cada uno iba desarrollando el código, y al finalizar se realizaron unas pruebas con conjuntos de entradas y sus posibles respuestas. Estas estarán registradas en un Excel adjunto al repositorio de la organización.

# 3- Organización

Nuestro principal medio de comunicación fue por Discord.

En primera instancia nos juntamos para analizar los requerimientos y poder complementarlos si es que los encontrábamos incompletos. Una vez terminada esta parte nos dividimos el trabajo donde:

- Matías realizó el CRUD de eventos y filtrado de estos, usando las Branch develop y filtrado.
- José realizó la autenticación y generación de reportes, usando la Branch auth\_rep.

Ambos intentamos seguir el flujo de trabajo de gitflow.

Mientras realizábamos los programas, cada uno probaba a medida que avanzaba de manera local. Si detectaba fallo, cada uno lo arreglaba

Primero José realizó el merge al main y luego Matías lo realizó.

Se dispuso de una reunión para la resolución del merge en main. Después se realizaron las últimas pruebas en conjunto.

## 4- Flujo de trabajo

Se creó un espacio de trabajo en Slack

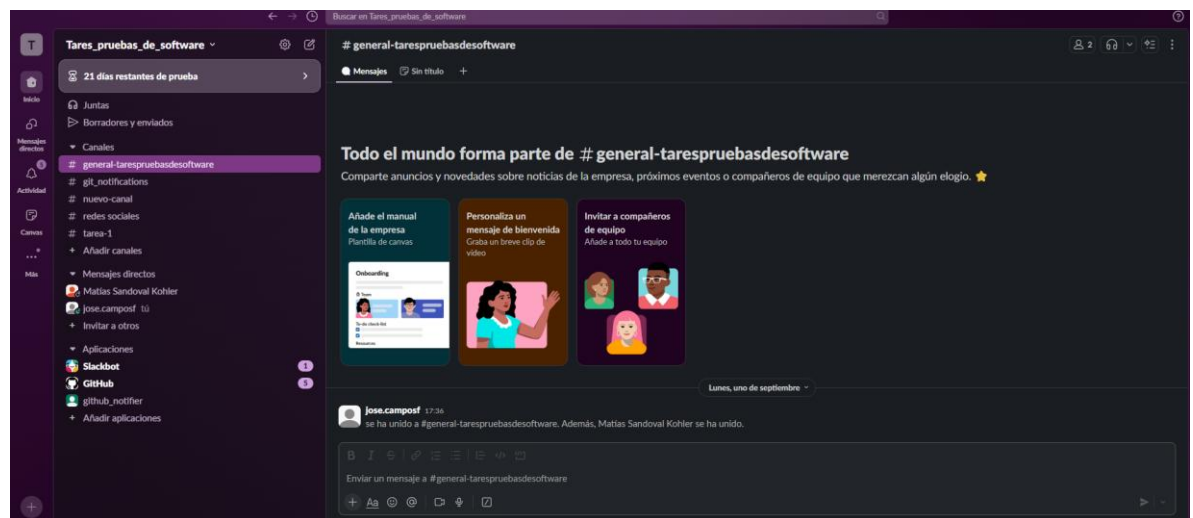


Figura 1: Slack Workplace

Después lo integramos con GitHub

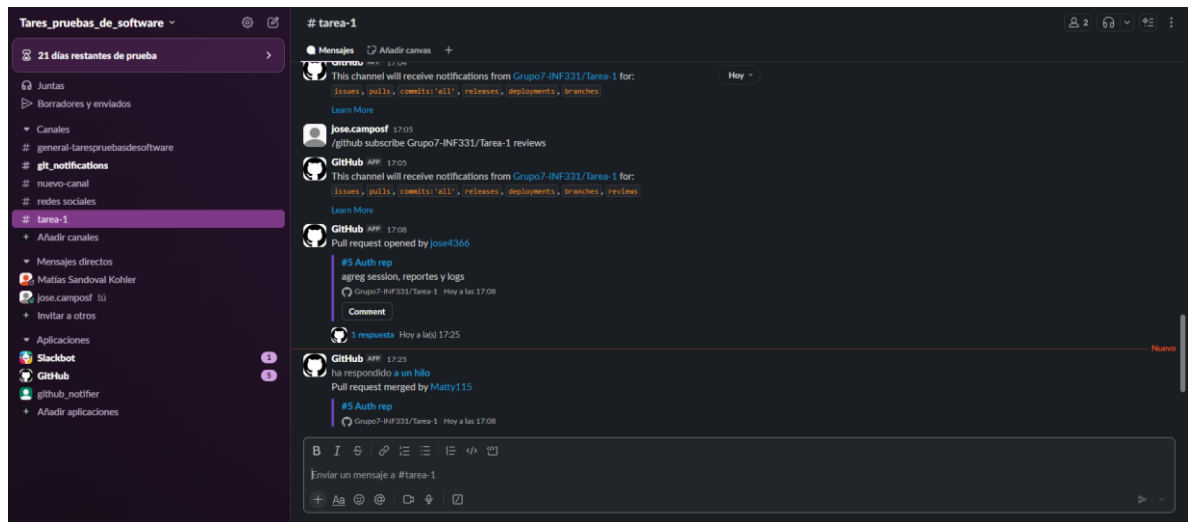


Figura 2: GitHub + Slack

Se utilizó gitflow. El main disponía de hacer pull request (protección de ramas) (ver figura 3), y la otra miembro tenía que revisarla y aceptar (Revisores/Aprobadores)(ver figura 4). La Branch develop y filtrado fueron usadas para realizar el CRUD y filtrado. Una vez realizado cada uno de las partes, se unió en la Branch main.

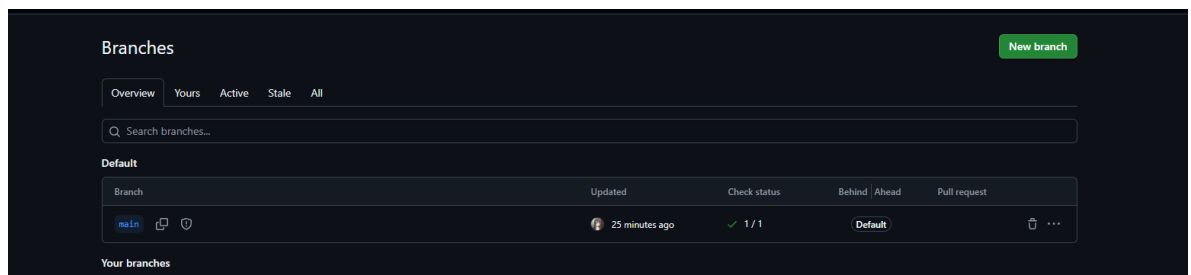


Figura 3: Protección de rama main

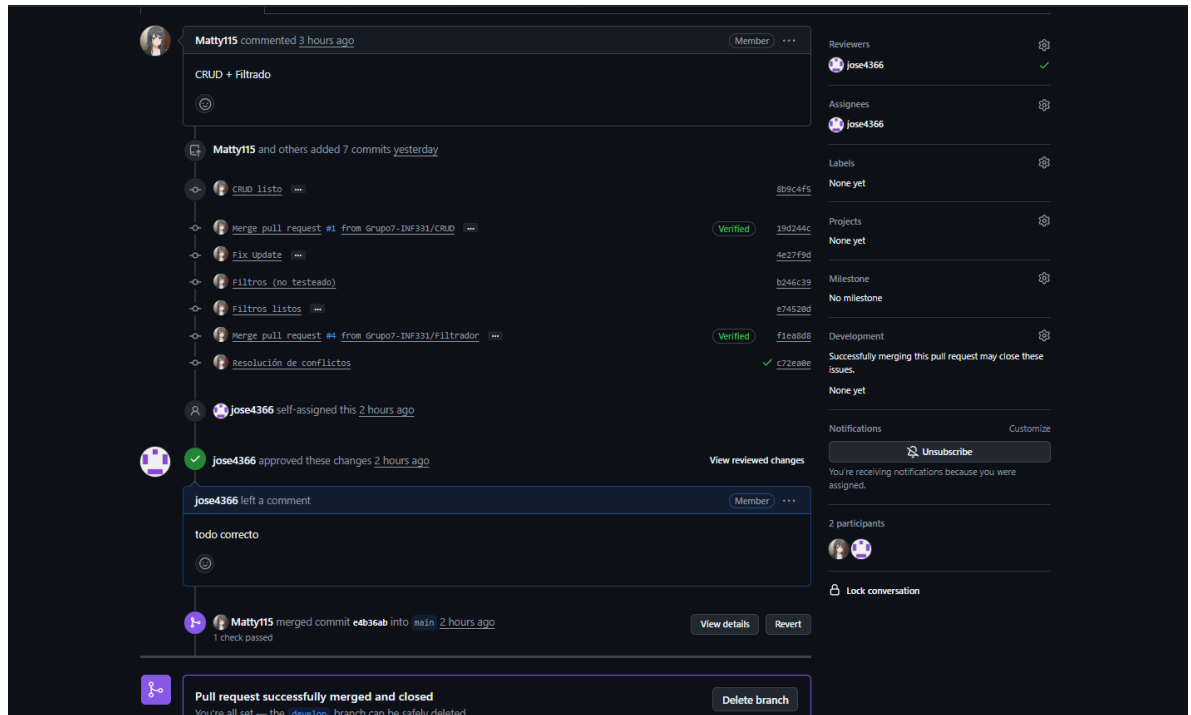


Figura 1: Revisores/Aprobadores

## 5- Problemas encontrados

Tuvimos un problema a la hora de usar GitHub + Slack, ya que al inicio invitamos al miembro a una sala, pero este solo mostraba los pull request, y no mostraba ni las branches ni todos los commits. Para resolver esto nos pusimos a investigar y había que pasarle al Github de slack un parámetro extra de Branch y commits (" /github subscribe Grupo7-INF331/Tarea-1 branches, /github subscribe Grupo7-INF331/Tarea-1 commits:all ").

Tuvimos un problema al hacer merge de ambas branches, y tuvimos que realizar la resolución de conflictos ambos en llamada diciendo que dejar en el código y que cambiar.